Resource Allocation in Visual Sensor Networks Using a Reinforcement Learning Framework

Katerina Pandremmenou, Nikolaos Tziortziotis, Lisimachos P. Kondi and Konstantinos Blekas
Department of Computer Science, University of Ioannina, GR-45110, Ioannina, Greece

***Abstract*—In recent years, video delivery over wireless visual sensor networks (VSNs) has gained increasing attention. The lossy compression and channel errors that occur during wireless multimedia transmissions can degrade the quality of the transmitted video sequences. This paper addresses the problem of cross-layer resource allocation among the nodes of a wireless direct-sequence code division multiple access (DS-CDMA) VSN. The optimal group of pictures (GoP) length during the encoding process is also considered, based on the motion level of each video sequence. Three optimization criteria that optimize a different objective function of the video qualities of the nodes are used. The nodes' transmission parameters, i.e., the source coding rates, channel coding rates and power levels can only take discrete values. In order to tackle the resulting optimization problem, a reinforcement learning (RL) strategy that promises efficient exploration and exploitation of the parameters' space is employed. This makes the proposed methodology usable in large or continuous state spaces as well as in an online mode. Experimental results highlight the efficiency of the proposed method.**

***Keywords*—Cross–layer optimization, group of pictures length, Markov decision processes, reinforcement learning, resource allocation, visual sensor network.**

I. Introduction

During the last years, wireless communications and networking have enjoyed huge commercial success thanks to advances in wireless technologies and industrial standards. The rapid growth of broadband wireless networks has enabled the development of wireless *visual sensor networks* (VSNs) [1]. VSNs support a plethora of potential applications, ranging from security to environmental monitoring, health care, and teleconference systems. Thus, video delivery over such networks has gained increasing attention, while considerable progress has been made in solving numerous wireless sensor networking challenges.

Nevertheless, a great concern in most VSNs' applications is to provide mechanisms able to guarantee high levels of *quality of service* (QoS) in the real-time delivery of multimedia content. The time-varying nature and error-prone environment of wireless networks, as opposed to the delay-sensitive and bandwidth-intensive real-time multimedia applications, poses the need for the optimal configuration of the wireless transmission system. Furthermore, the errors that occur during wireless multimedia transmissions, in conjunction with the lossy source coding techniques, deteriorate the quality of the video sequences at the decoder. Thus, careful treatment is also required during video encoding in order to acquire high coding performance and robustness to transmission errors.

In our previous works [2], [3], we assumed a wireless VSN, where an application-driven cross-layer optimization scheme was proposed for the dynamic adjustment of the sensor nodes' transmission parameters across all network layers. Such a scheme provides the opportunity for increased network resource usage and user profit maximization, at the same time. A literature review demonstrates that, whereas joint source and channel coding as well as energy consumption minimization have been the main objectives in wireless VSN research [2], [3], [4], [5], [6], little evidence is available for the investigation of efficient coding techniques by applying adaptive *group of pictures* (GoP) length, at the same time. This latter approach aims at the enhancement of video resiliency to channel errors during wireless transmissions. The works presented in [7] and [8] propose GoP structures adaptive to video content, without addressing resource allocation issues, at the same time.

The H.264/AVC video coding standard defines three frame types for video coding: intra frames (IDR, I), predictive frames (P) and bidirectionally predictive frames (B). Intra frames are coded without reference to other frames, while the difference between P-frames and B-frames is the number of reference frames they are allowed to use for coding. An *instantaneous decoding refresh* (IDR) frame is a regular I-frame with the constraint that pictures appearing after it in the bitstream cannot use the pictures appearing before it as references. A GoP, which is a group of successive pictures within a coded video stream, always begins with an IDR-frame, and therefore the propagation of any errors within the GoP structure is stopped by the next IDR-frame.

The aim of this paper is twofold. Firstly, it studies the cross-layer resource allocation problem among the nodes of a wireless VSN and secondly, it deals with the optimal IDR-frame placement during the encoding process, based on the motion level included in each video sequence. The resource allocation issue is tackled using three optimization schemes. The first one minimizes the average video distortion of all nodes [2], the second one is the *Nash bargaining solution* (NBS) [3] and the last scheme maximizes the sum of all nodes' utilities.

In our study, we have to deal with a discrete optimization problem, since all nodes' transmission parameters, i.e., source coding rates, channel coding rates and power levels can only take discrete values. Discrete optimization problems were also resolved in our previous works [2], [3]. However, in the current work, each node can select among a larger number of possible values for the power levels. Furthermore, an extra element of the present study is the assumption about four different levels of motion in the scenes captured by the nodes, as opposed to the coarser approach of only two motion levels of our previously published works. Combining these two considerations about more possible choices for the

978-1-4673-5807-1/13/$31.00 ©2013 IEEE

nodes' power levels and more levels of motion included in the scenes captured by the nodes, it is clear that the problem's dimensionality significantly increases, rendering the use of the brute-force search algorithm rather impractical. Hence, the current work abandons the traditional *exhaustive search* (ES) algorithm used in [2], [3], and enjoys the benefits of other innovative optimization methods extracted from the area of *reinforcement learning* (RL) [9].

RL provides an elegant framework for making decisions under uncertainty based on the maximization of the expected utility functions. A significant contribution of this work is the incorporation of an RL scheme in the resource allocation problem, which allows the controller to make optimal decisions in unknown environments with very large or continuous state spaces. RL has been used extensively in control strategies for video quality processing [10], surpassing a lot of difficulties in the particular field. More specifically, RL discovers an optimal or near-optimal policy in the early stages of the learning process, while at the same time it is able to adapt to potential changes of the environment. The benefit of the latter feature clearly emerges in the online case, where the environment changes dynamically over time.

In this work, we use the tabular SARSA algorithm which is a model-free on-policy algorithm that belongs in the family of *temporal difference* (TD) algorithms [9]. The resource allocation problem examined in the present paper is modeled appropriately as a *Markov decision process* (MDP) [11]. The particular approach exploits the received raw experience, discovering the optimal combination of the nodes' transmission parameters in a more efficient way. Roughly speaking, SARSA constructs a map that allows us to explore the best parameters with the minimum effort, starting from any randomly selected parameters' combination. Last but not least, the specific RL approach gives the opportunity for the proposed scheme to be used in an online mode.

The rest of the paper is organized as follows: Section II describes the basic architecture of the considered wireless DS-CDMA VSN. Section III presents the video distortion model, accounts for the necessity for efficient GoP structures and describes how the adaptive GoP structure is applied in this work. In Section IV, the employed resource allocation schemes are presented and Section V describes the proposed RL approach for tackling the discrete optimization problem of this study. Experimental results are provided in Section VI and conclusions are drawn in Section VII.

II. Visual Sensor Network

VSNs are networks of wireless, interconnected smart devices, the so-called sensors, each of which is equipped with a video camera that enables capturing visual data. Apart from the data acquisition, sensor nodes are also capable of processing multimedia streams in real-time, since they have some local image processing and communication capabilities. The *centralized control unit* (CCU), which lies at the network layer, collects the data from the wireless transceivers and transmits information to the nodes in order to request changes in their transmission parameters, i.e., the source coding rates, channel coding rates, and power levels that are taking place at the application layer, data link layer, and physical layer, respectively.

Direct-sequence code division multiple access (DS-CDMA) is the channel access method considered in the current study. DS-CDMA systems are usually interference-limited and, thus, it is common for the thermal and background noise to be neglected. The power level for each node, k, is given by $S_k = E_k R_k$; E_k is the energy per bit and R_k is the total bit rate. It equals $R_k = R_{\mathrm{s},k}/R_{\mathrm{c},k}$, where $R_{\mathrm{s},k}$ represents the source coding rate and $R_{\mathrm{c},k}$ the channel coding rate. Since the unit measure for the $R_{\mathrm{s},k}$ is bits per second (bps), and taking into account that $R_{\mathrm{c},k}$ is a dimensionless number, it follows that R_k is also measured in bps.

Low-motion video sequences can be encoded using a lower source coding rate. Thus, given that a target bit rate constraint is imposed on every node of the wireless VSN, low-motion nodes can use a larger bit rate for channel coding. Consequently, the power level required for the transmission of low motion scenes is kept at low levels. Since DS-CDMA allows all nodes to transmit over the same channel, transmissions of one node cause interference to the transmissions of the other nodes. Moreover, considering that the nodes are battery-operated, the need for power conservation is imperative. On the other hand, power levels should be adequately high to permit data transmissions and maintain the quality of the video reception.

In our investigation, we assume that interference can be approximated by *additive white Gaussian noise* (AWGN) [6]. Thus, the energy per bit to *multiple access interference* (MAI) ratio is given by:

$$\frac{E_k}{N_0} = \frac{\frac{S_k}{R_k}}{\sum_{j\neq k}^{K} \frac{S_j}{W_t}}, \qquad k = 1, 2, \ldots, K, \tag{1}$$

where $N_0/2$ is the two-sided noise power spectral density due to MAI and W_t is the total available bandwidth. The index k refers to the corresponding node and j to each interfering node.

For the channel coding, we assume *rate compatible punctured convolutional* (RCPC) codes [12], which allow the use of Viterbi's upper bounds on the bit error probability P_{b}. Assuming *binary phase shift keying* (BPSK) as the employed modulation scheme, P_{b} satisfies the inequality $P_{\mathrm{b}} \leqslant \frac{1}{P}\sum_{d=d_{\mathrm{free}}}^{\infty} c_d P_d$, where $P_d = \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{dR_{\mathrm{c}}E_k}{N_0}}\right)$. The parameter P is the period of the code, d_{free} is the free distance of the code and c_d is the information error weight. The complementary error function is denoted as $\operatorname{erfc}()$, while R_{c} is the channel coding rate.

III. Video Coding and Transmission

The quality of a video sequence is highly related with the lossy compression techniques applied at the encoder as well as with the number of occurred errors during wireless data transmissions. Consequently, in order to achieve the maximization of the network QoS, in this paper, we focus on both these problem aspects.

A. Video Distortion Model

Our previous experience with *universal rate distortion characteristics* (URDCs) [2], [3], has shown that it is an efficient tool that can be used to express the expected distortion

$E[D_{\text{s+c},k}]$ of node k, as a function of the bit error probability (bit error rate), P_{b}, after channel decoding. Similarly, in this paper, we make use of URDCs given by the model:

$$E[D_{\text{s+c},k}] = \alpha \left[\log_{10}\left(\frac{1}{P_{\text{b}}}\right)\right]^{-\beta}, \quad (2)$$

where α and β are two positive parameters. In the end of the next subsection, we explain how the parameters α and β are obtained.

B. GoP Structure

The H.264/AVC codec has the flexibility to determine the frequency of IDR-frames on the encoding side. Since IDR-frames are independently coded frames, the errors that occur within a GoP propagate to the following frames until the next IDR-frame is found. Generally, the more IDR-frames are included in a video stream, the more editable it is and the greater its size is. Since predictive coding techniques are applied during encoding, the effect of channel errors on the video can have a tremendous impact after video transmission over error-prone environments. Thus, it is important to apply techniques that ensure a tolerable level of QoS.

It is widely accepted that scene changes or large variations can happen at any location in a video stream. This means that it is important to consider the video content in order to wisely arrange each of the IDR-, P- and B-frames in a GoP. Clearly, in the beginning of a new scene or after an abrupt scene change, an IDR-frame insertion is required in order to prohibit poor prediction for the next frames, since this type of intra frames does not allow the following frames to use frames appearing before it as references. Alternatively, when low levels of motion are included in a video stream, it is more efficient to use more P- or B-frames, instead of IDR-frames, to enhance video coding performance.

In this work, we experiment on the GoP length, i.e., the distance between two consecutive IDR-frames, during the encoding process of the video sequences. We assume that the nodes of the considered VSN record four different levels of motion: low, low-medium, medium-high and high. Thus, they are clustered in four motion classes, based on the motion level included in the scenes they record. Figure 1 presents the different motion levels represented by each considered video sequence.

Each of the video sequences is compressed using four different GoP lengths, at three different source coding rates. We simulate video transmission through the channel by dropping packets from the video streams, at three different bit error rates. Due to the fact that channel errors and packet drops occur randomly, the video distortion attributed to lossy compression and channel errors is a random variable. In light of this, the video distortion is averaged over a number of independent experiments. The parameters α and β of Eq. (2) are then determined through a mean-squared-error optimization procedure, using a small number of $(E[D_{\text{s+c},k}], P_{\text{b}})$ pairs.

Hence, each motion class has its own set of α and β parameters, which also depend on the source-channel coding rate and GoP length. Having compressed each video bitstream using four different GoP lengths, we test all possible GoP length combinations of all video bitstreams. Each different combination results in different values for the α and β parameters. For each (α,β) pairwise values, we run the optimization procedure using each of the schemes described in the next section. Those (α,β) values that satisfy the objective of each scheme are chosen as optimal. The GoP length for each motion class that produces the optimal (α,β) values is proven to be the most efficient one, since it leads to the ultimate video quality enhancement.

Fig. 1. Motion level represented by each video sequence.

IV. Resource Allocation Schemes

Given that the K nodes of the VSN are grouped into M motion classes, we substitute Eq. (1) into P_d's equation, P_d into P_{b}'s equation, and finally P_{b} into Eq. (2). Then, it follows that the expected distortion $E[D_{\text{s+c},cl}]$ for the motion class cl is a function of the source coding rate, for class cl, channel coding rate, for class cl, and power levels of all motion classes.

Before the presentation of the resource allocation schemes, it is necessary to define the utility function. The *utility function*, U_{cl}, constitutes a measure of relative satisfaction for motion class cl. In our problem, it is defined equivalently to the *peak signal to noise ratio* (PSNR) [3], i.e., $U_{cl} = 10\log_{10}(255^2/E[D_{\text{s+c},cl}])$, and is measured in decibel (dB). The quantity $E[D_{\text{s+c},cl}]$ represents the expected distortion given by Eq. (2). The larger the value of the utility function, the better the video quality for motion class cl, and vice versa.

In the following, we summarize the optimization criteria used in order to determine the nodes' transmission parameters.

1) **Minimum Average Distortion** (MAD)
This criterion aims at the minimization of the average distortion of the M motion classes of the network:

$$\min \frac{1}{M}\sum_{cl=1}^{M} E[D_{\text{s+c},cl}](R_{\text{s},cl}, R_{\text{c},cl}, S), \quad (3)$$

while it does not assert fairness among the M motion classes. Hence, distortion is allowed to vary significantly among the motion classes as long as the average distortion is kept to minimal levels.

2) **Nash Bargaining Solution** (NBS)
This bargaining solution, based on its fairness axioms [3], can be determined as:

$$\max_{U \geq dp} \prod_{cl=1}^{M} (U_{cl}(R_{\text{s},cl}, R_{\text{c},cl}, S) - dp_{cl})^{a_{cl}}, \quad \sum_{cl=1}^{M} a_{cl} = 1. \quad (4)$$

The vector U includes the utilities of the M motion classes and the vector dp is the *disagreement point*, which includes the minimum utilities that each motion class expects by joining the game, without cooperating with the other classes. In the present work, the disagreement point is imposed by the designer of the system. The

amount a_{cl} corresponds to the bargaining power assigned to each motion class cl and declares the advantage of that class in the resource allocation game. The larger the value of the bargaining power is, the more advantaged the motion class is, and vice versa. In our implementation, we assumed that all K nodes of the network are equally advantaged. Thus, given the node clustering into M motion classes, the bargaining power assigned to each motion class cl is proportional to its cardinality N_{cl}. Therefore, $a_{cl} = N_{cl}/K$.

3) **Maximum Total Utility** (MTU)
In some cases, all the nodes of the network aspire to maximize the total system utility. Therefore, assuming again a node clustering into M motion classes, we have to maximize the function:

$$\max \sum_{cl=1}^{M} U_{cl}(R_{\mathrm{s},cl}, R_{\mathrm{c},cl}, S), \tag{5}$$

where U_{cl} corresponds to the utility of the motion class cl.

V. Resource Allocation using Reinforcement Learning

At this point, we introduce the formulation of the resource allocation problem as a *Markov decision process* (MDP) [11] and we also present the *reinforcement learning* (RL) scheme, which is incorporated in the controller, i.e., the CCU. The resource allocation problem considered in this study is treated as a discrete optimization problem (discrete nodes' transmission parameters). Although in the past this problem had been encountered using the heuristic optimization methodology of *exhaustive search* (ES) [2], [3], this is not feasible in the specific work. In our case, the controller has to select among a considerably larger set of possible variable combinations compared with the previous works. Nevertheless, the major handicap of the ES algorithm is its computational complexity, which renders its use prohibitive in the online mode.

According to our proposed methodology, the learning optimization problem is formulated in a sequential decision framework and is modeled as an MDP [11]. Roughly speaking, an MDP involves a decision agent (controller) that repeatedly observes the current state of the controlled system, takes a decision among the ones allowed in that state, and then observes a new state as well as a reward that will drive its future decisions. The MDP is typically denoted as a tuple $\{\mathcal{X}, \mathcal{U}, \mathcal{R}, \mathcal{P}, \gamma\}$, where $\mathcal{X}$ and $\mathcal{U}$ are the state and action spaces, respectively; $\mathcal{R}$ is the reward function that specifies the importance of each transition; $\mathcal{P}$ is the state transition distribution; and $\gamma \in [0, 1]$ is the discount factor that determines the importance of the future rewards.

In the learning problem of resource allocation studied in this paper, we consider the state space as the Cartesian product of eight sets;

$$\mathcal{X} \triangleq C_1 \times C_2 \times C_3 \times C_4 \times S_1 \times S_2 \times S_3 \times S_4.$$

In this way, a state is represented as an eight-dimensional vector. Each of the first four variables denotes the source–channel coding rate combination for the motion class cl ($C_{cl} \in \{1, 2, 3\}$) and each of the remaining variables denotes the power level for the motion class cl, ($S_{cl} \in \{5, 7, 9, 11, 13, 15\}$), $cl = \{1, \ldots, 4\}$. Moreover, the action space consists of 17 actions, two for each dimension. At each time step, the controller can increase or decrease one of the state variables. Additionally, we give the ability to the controller to leave the state variables unchanged, by remaining at the same state. Regarding the reward function, it specifies the gain obtained during a transition from the current state $\boldsymbol{x}$ to the next state $\boldsymbol{x'}$, as given by the difference between the values of the objective functions corresponding to the specific states.

A stationary policy $\pi : \mathcal{X} \rightarrow \mathcal{U}$ is a mapping from states to actions and denotes a mechanism for choosing actions appropriately. The notion of *value function* is of central interest in RL tasks. Given a policy π, the value $V^{\pi}(\boldsymbol{x})$ of a state $\boldsymbol{x}$ is defined as the expected discounted sum of rewards, obtained, starting from this state and following the policy:

$$V^{\pi}(\boldsymbol{x}) = E_{\pi}\left[\mathcal{R}(\boldsymbol{x}_t) + \gamma V^{\pi}(\boldsymbol{x}_{t+1}) | \boldsymbol{x}_t = \boldsymbol{x}\right]. \tag{6}$$

This is actually a Bellman equation, which expresses a relationship between the value of a state and the values of its successor states. Similarly, the state-action value function $Q(\boldsymbol{x}, u)$ denotes the expected cumulative reward as received by taking action u in state $\boldsymbol{x}$, and following policy π:

$$Q^{\pi}(\boldsymbol{x}, u) = E_{\pi}\left[\mathcal{R}(\boldsymbol{x}_t) + \gamma V^{\pi}(\boldsymbol{x}_{t+1}) | \boldsymbol{x}_t = \boldsymbol{x}, u_t = u\right]. \tag{7}$$

The objective of an RL task is to estimate an optimal policy π^* by choosing actions that yield the optimal state-action value function: $\pi^*(\boldsymbol{x}) = \arg\max_u Q^*(\boldsymbol{x}, u)$.

The *temporal difference* (TD) family of algorithms [9] provides an elegant framework for solving prediction problems. The main advantage of this class of algorithms is its ability to learn directly from raw experience, without any further information. One of the most popular TD algorithms is the SARSA algorithm [13], which is a *bootstrapping* technique. More specifically, this is an on-policy control method, which is based on the state-action value function estimation. The predicted Q value of the new visited state-action pair and the received reward are used to calculate an improved estimate for the Q value of the previous visited state-action pair:

$$\delta_t = r_t + \gamma Q(\boldsymbol{x}_{t+1}, u_{t+1}) - Q(\boldsymbol{x}_t, u_t). \tag{8}$$

The above quantity is known as the one-step TD error and is used for adjusting the weights of the policy, by performing a stochastic gradient descent scheme:

$$Q(\boldsymbol{x}_t, u_t) \leftarrow Q(\boldsymbol{x}_t, u_t) + \eta \delta_t, \tag{9}$$

where the parameter η is the learning rate that controls the update rule. Moreover, we can combine the SARSA algorithm with the *eligibility traces*, SARSA(λ) [13], allowing the update rule to propagate the TD error backward over the current trajectory of states. It has been proven that TD algorithms are able to find the optimal policy with probability 1 [14]. This fact gives us the opportunity to find the optimal variable combination with certainty, starting from each initial state and following the learned policy.

VI. Experimental Results

In this work, the $K = 100$ nodes of the network were clustered into $M = 4$ motion classes, with each class consisting of 25 nodes. All video sequences (Fig. 1) were at *quarter common intermediate format* (QCIF) resolution and the H.264/AVC High profile for 4:2:0 color format video was selected to compress each of them. The RCPC codes had mother rate $1/4$ [12], the total bandwidth was $W_t = 20$ MHz and the target bit rate $R_k = 96$ kbps. The tested GoP lengths were $\{3, 5, 10, 30\}$. The set of admissible source and channel coding rate combinations was $\mathbf{C} \in \{1 : (32, 1/3), 2 : (48, 1/2), 3 : (64, 2/3)\}$ and the power levels assumed values from the set $\mathbf{S} = \{5, 7, 9, 11, 13, 15\}$ mW. The disagreement point was $dp = (25, 25, 25, 25)^\top$ dB. In order to encourage exploration in the adopted RL scheme, the initial state-action value functions were selected optimistically [9]. The specific optimization problem was treated as a continuous task, where the optimal solution was reached, when the controller remained in the same state for a maximum number of steps (stopping criterion).

In the following, Table I presents the optimal determination of the transmission parameters for all considered criteria. Although all possible combinations for the GoP length for each video sequence were tested, due to lack of space we cite only three cases. Case 1: all video sequences are compressed with GoP length 30 (relatively infrequent IDR-frame placement), Case 2: all video sequences are compressed with GoP length 3 (relatively frequent IDR-frame placement) and Case 3: each motion class selects the optimal GoP length.

Let index 1 denote the high motion class, index 2 the medium-high motion class, index 3 the low-medium motion class and index 4 the low motion class. Thus, GoP_{cl}, S_{cl} and C_{cl} refer to the GoP, power level and source-channel coding rate combination for the class cl, $cl = \{1, \ldots, 4\}$. From the obtained results, we observe that when the optimal GoP length is selected for each motion class, we receive an increase in the total PSNR (sum of the PSNRs) of all motion classes up to 4.2 dB compared to the case when GoP length is 30 and up to 9.6 dB when GoP length is 3.

Furthermore, as Fig. 2 shows, for the MAD and NBS criteria, when optimal GoP length is selected, all video sequences increase their own utilities compared to the other two GoP length considerations. For the MTU criterion, only the "Foreman" video sequence augments its utility compared to the other two GoP length considerations. However, the total PSNR increase achieved using the optimal GoP length is 7.8 dB compared to Case 2, and 3.7 dB compared to Case 1, which is a considerable PSNR increase.

In the following, Fig. 3 compares the PSNR values achieved by each considered criterion, for all tested video sequences. The MAD favors the video sequences including high and medium-high amounts of motion, while the MTU is preferred by the nodes that capture low and low-medium amounts of motion. Regarding the NBS, it is the criterion that presents the smallest discrepancy between the PSNR values of all video sequences, being a compromise between the values of MAD and MTU, for all video sequences.

Last but not least, Fig. 4 depicts the mean number of steps that the SARSA algorithm requires compared to the ES algorithm. It is obvious that SARSA needs a significantly smaller number of steps and hence less time, in order to discover the optimal combination of nodes' transmission parameters, for all considered criteria. This is attributed to the efficient way that the particular algorithm uses the received information from the environment. These two approaches, i.e., SARSA and ES, will become non comparable in the case of online processing.

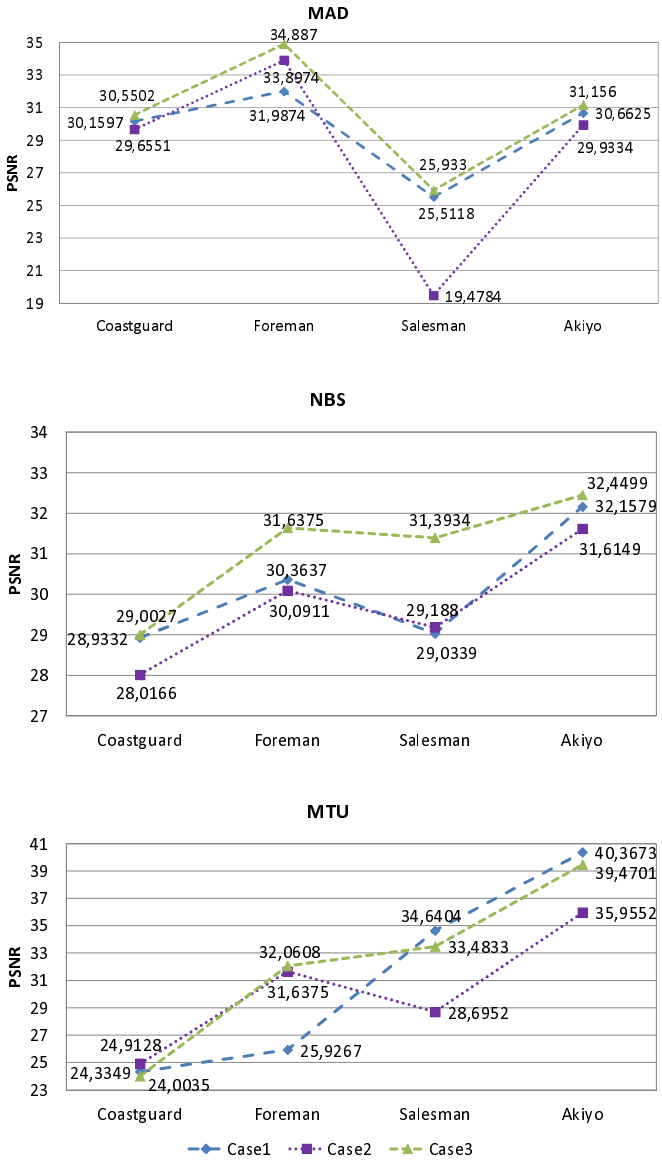

Fig. 2. PSNR achieved for all video sequences for 3 different GoP lengths.

VII. Conclusions

This work dealt with the problem of cross-layer resource allocation among the nodes of a wireless DS-CDMA VSN. Additionally, the optimal GoP structure for the encoding of each video sequence captured by the nodes was the other main objective of this paper. For the determination of the nodes' transmission parameters, three optimization criteria were used. Allowing the nodes to select among various GoP lengths for the encoding of the video they capture, considering the motion

TABLE I. RESOURCE ALLOCATION FOR ALL CONSIDERED CRITERIA.

MAD													
Case	GoP_1	GoP_2	GoP_3	GoP_4	S_1	S_2	S_3	S_4	C_1	C_2	C_3	C_4	Total PSNR
1	30	30	30	30	15	15	5	5	3	3	1	1	118.3214
2	3	3	3	3	15	11	5	5	1	1	1	1	112.9643
3	30	3	30	30	15	13	5	5	3	1	1	1	122.5263
NBS													
Case	GoP_1	GoP_2	GoP_3	GoP_4	S_1	S_2	S_3	S_4	C_1	C_2	C_3	C_4	Total PSNR
1	30	30	30	30	11	11	7	5	3	3	1	1	120.4887
2	3	3	3	3	15	9	13	7	1	1	1	1	118.9107
3	30	3	30	30	15	11	13	7	3	1	2	1	124.4835
MTU													
Case	GoP_1	GoP_2	GoP_3	GoP_4	S_1	S_2	S_3	S_4	C_1	C_2	C_3	C_4	Total PSNR
1	30	30	30	30	5	7	15	13	1	1	3	3	125.2693
2	3	3	3	3	11	11	13	11	1	1	1	1	121.2006
3	30	3	30	30	5	11	15	13	1	1	3	3	129.0177

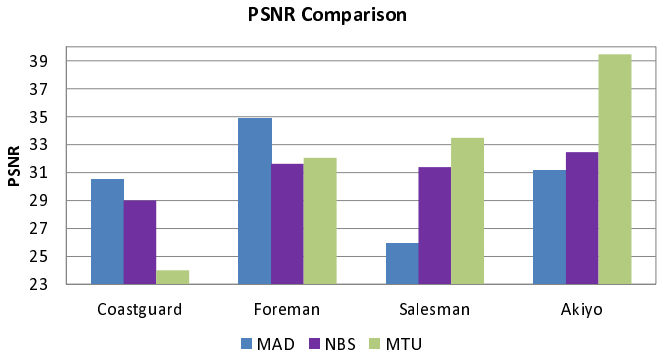

Fig. 3. PSNR achieved by each criterion for all video sequences.

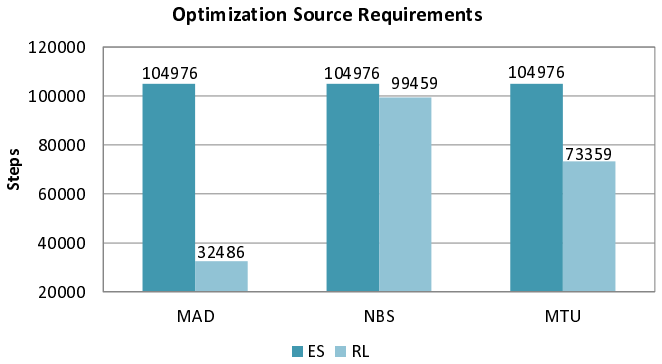

Fig. 4. Steps required by ES and RL to reach to the solution.

level included in those scenes, video quality enhancement was observed compared to fixed GoP length considerations. Furthermore, the RL approach adopted in this paper to tackle the resulting optimization problem was proven extremely efficient compared to the brute-force search approach. Although both ES and SARSA algorithms are able to reach to the optimal solution, SARSA requires far less steps, making the proposed methodology applicable in online form.

ACKNOWLEDGEMENT

Effort sponsored by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number $FA8655-12-1-0001$. The U.S Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright notation thereon.

REFERENCES

[1] I. Akyildiz, T. Melodia, and K. Chowdury, "Wireless multimedia sensor networks: A survey," *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 32–39, Dec. 2007.

[2] E. S. Bentley, L. P. Kondi, J. D. Matyjas, M. J. Medley, and B. W. Suter, "Spread spectrum visual sensor network resource management using an end–to–end cross–layer design," *IEEE Transactions on Multimedia*, vol. 13, no. 1, pp. 125–131, Feb. 2011.

[3] L. P. Kondi and E. S. Bentley, "Game–theory–based cross–layer optimization for wireless DS–CDMA visual sensor networks," in *17th IEEE International Conference on Image Processing (ICIP)*, Sept. 2010, pp. 4485–4488.

[4] J. Yuan and W. Yu, "Joint source coding, routing and power allocation in wireless sensor networks," *IEEE Transactions on Communications*, vol. 56, no. 6, pp. 886–896, 2008.

[5] W. Wang, D. Peng, H. Wang, H. Sharif, and H.-H. Chen, "Energy–constrained distortion reduction optimization for wavelet–based coded image transmission in wireless sensor networks," *IEEE Transactions on Multimedia*, vol. 10, no. 6, pp. 1169–1180, Oct. 2008.

[6] Y. S. Chan and J. W. Modestino, "A joint source coding–power control approach for video transmission over CDMA networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 10, pp. 1516–1525, 2006.

[7] B. Zatt, M. Porto, J. Scharcanski, and S. Bampi, "GoP structure adaptive to the video content for efficient H.264/AVC encoding," in *17th IEEE International Conference on Image Processing (ICIP)*, Sept. 2010, pp. 3053–3056.

[8] J. Lee, I. Shin, and H. Park, "Adaptive intra–frame assignment and bit–rate estimation for variable GoP length in H.264," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 10, pp. 1271–1279, Oct. 2006.

[9] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*. MIT Press Cambridge, USA, 1998.

[10] C. C. Wüst, L. Steffens, W. F. Verhaegh, R. J. Bril, and C. Hentschel, "QoS control strategies for high–quality video processing," *Real–Time Syst.*, no. 1–2, May 2005.

[11] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[12] J. Hagenauer, "Rate–compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 389–400, 1988.

[13] G. A. Rummery and M. Niranjan, "On–line Q–learning using connectionist systems," Tech. Rep., 1994.

[14] P. Dayan, "The convergence of TD(λ) for general lambda," *Machine Learning*, vol. 8, pp. 341–362, 1992.